Introduction

5 questions

1 point

1.
A computer program is said to learn from experience E with respect to some task T and some performance measure P if its performance on T, as measured by P, improves with experience E. Suppose we feed a learning algorithm a lot of historical weather data, and have it learn to predict weather. In this setting, what is T?

- The probability of it correctly predicting a future date's weather.
- The weather prediction task.
- None of these.
- The process of the algorithm examining a large amount of historical weather data.

1 point

2.

Suppose you are working on weather prediction, and use a

learning algorithm to predict tomorrow's temperature (in

degrees Centigrade/Fahrenheit).

Would you treat this as a classification or a regression problem?

- ○ Regression
- ○ Classification

---

1 point

3.

Suppose you are working on stock market prediction, Typically

tens of millions of shares of Microsoft stock are traded

(i.e., bought/sold) each day. You would like to predict the

number of Microsoft shares that will be traded tomorrow.

Would you treat this as a classification or a regression problem?

- ○ Classification
- ○ Regression

---

1 point

4.

Some of the problems below are best addressed using a supervised learning algorithm, and the others with an unsupervised learning algorithm. Which of the following would you apply supervised learning to? (Select all that apply.) In each case, assume some appropriate dataset is available for your algorithm to learn from.

- ☐ Given genetic (DNA) data from a person, predict the odds of him/her developing diabetes over the next 10 years.

- [ ] Examine the statistics of two football teams, and predicting which team will win tomorrow's match (given historical data of teams' wins/losses to learn from).

- [ ] Take a collection of 1000 essays written on the US Economy, and find a way to automatically group these essays into a small number of groups of essays that are somehow "similar" or "related".

- [ ] Examine a large collection of emails that are known to be spam email, to discover if there are sub-types of spam mail.

---

1 point

5.
Which of these is a reasonable definition of machine learning?

- ○ Machine learning is the science of programming computers.
- ○ Machine learning learns from labeled data.
- ○ Machine learning is the field of allowing robots to act intelligently.
- ○ Machine learning is the field of study that gives computers the ability to learn without being explicitly programmed.

---

4 questions unanswered

Submit Quiz

Linear Regression with One Variable

5 questions

1 point

1.
Consider the problem of predicting how well a student does in her second year of college/university, given how well they did in their first year.

Specifically, let x be equal to the number of "A" grades (including A-. A and A+ grades) that a student receives in their first year of college (freshmen year). We would like to predict the value of y, which we define as the number of "A" grades they get in their second year (sophomore year).

Questions 1 through 4 will use the following training set of a small sample of different students' performances. Here each row is one training example. Recall that in linear regression, our hypothesis is $h_\theta(x) = \theta_0 + \theta_1 x$, and we use m to denote the number of training examples.

x	y
5	4
3	4
0	1
4	3

For the training set given above, what is the value of m? In the box below, please enter your answer (which should be a number between 0 and 10).

Enter answer here

1 point

2.
Consider the following training set of $m = 4$ training examples:

x	y
1	0.5
2	1
4	2
0	0

Consider the linear regression model $h_\theta(x) = \theta_0 + \theta_1 x$. What are the values of θ_0 and θ_1 that you would expect to obtain upon running gradient descent on this model? (Linear regression will be able to fit this data perfectly.)

- $\theta_0 = 0, \theta_1 = 0.5$
- $\theta_0 = 1, \theta_1 = 1$
- $\theta_0 = 0.5, \theta_1 = 0.5$
- $\theta_0 = 1, \theta_1 = 0.5$
- $\theta_0 = 0.5, \theta_1 = 0$

1 point

3. Suppose we set $\theta_0 = -1, \theta_1 = 0.5$. What is $h_\theta(4)$?

Enter answer here

1 point

4.

Let f be some function so that

$f(\theta_0, \theta_1)$ outputs a number. For this problem,

f is some arbitrary/unknown smooth function (not necessarily the

cost function of linear regression, so f may have local optima).

Suppose we use gradient descent to try to minimize $f(\theta_0, \theta_1)$

as a function of θ_0 and θ_1. Which of the

following statements are true? (Check all that apply.)

- ☐ If θ_0 and θ_1 are initialized at

  a local minimum, then one iteration will not change their values.

- ☐ Even if the learning rate α is very large, every iteration of

  gradient descent will decrease the value of $f(\theta_0, \theta_1)$.

- ☐ If the learning rate is too small, then gradient descent may take a very long

  time to converge.

- ☐ If θ_0 and θ_1 are initialized so that $\theta_0 = \theta_1$, then by symmetry (because we do simultaneous updates to the two parameters), after one iteration of gradient descent, we will still have $\theta_0 = \theta_1$.

---

1 point

5.

Suppose that for some linear regression problem (say, predicting housing prices as in the lecture), we

have some training set, and for our training set we managed to find some θ_0, θ_1 such that $J(\theta_0, \theta_1) = 0$. Which

of the statements below must then be true? (Check all that apply.)

☐ This is not possible: By the definition of $J(\theta_0, \theta_1)$, it is not possible for there to exist

θ_0 and θ_1 so that $J(\theta_0, \theta_1) = 0$

☐ For these values of θ_0 and θ_1 that satisfy $J(\theta_0, \theta_1) = 0$,

we have that $h_\theta(x^{(i)}) = y^{(i)}$ for every training example $(x^{(i)}, y^{(i)})$

☐ We can perfectly predict the value of y even for new examples that we have not yet seen.

(e.g., we can perfectly predict prices of even new houses that we have not yet seen.)

☐ For this to be true, we must have $\theta_0 = 0$ and $\theta_1 = 0$

so that $h_\theta(x) = 0$

---

3 questions unanswered

Submit Quiz

Linear Regression with Multiple Variables

5 questions

1 point

1.

Suppose *m*=4 students have taken some class, and the class had a midterm exam and a final exam. You have collected a dataset of their scores on the two exams, which is as follows:

midterm exam	(midterm exam)^2	final exam
89	7921	96
72	5184	74
94	8836	87
69	4761	78

You'd like to use polynomial regression to predict a student's final exam score from their midterm exam score. Concretely, suppose you want to fit a model of the form $h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2$, where x_1 is the midterm score and x_2 is (midterm score)^2. Further, you plan to use both feature scaling (dividing by the "max-min", or range, of a feature) and mean normalization.

What is the normalized feature $x_1^{(1)}$? (Hint: midterm = 89, final = 96 is training example 1.) Please round off your answer to two decimal places and enter in the text box below.

Enter answer here

1 point

2.
You run gradient descent for 15 iterations

with $\alpha = 0.3$ and compute

$J(\theta)$ after each iteration. You find that the

value of $J(\theta)$ **decreases slowly** and is still

decreasing after 15 iterations. Based on this, which of the

following conclusions seems most plausible?

- Rather than use the current value of α, it'd be more promising to try a larger value of α (say $\alpha = 1.0$).
- Rather than use the current value of α, it'd be more promising to try a smaller value of α (say $\alpha = 0.1$).
- $\alpha = 0.3$ is an effective choice of learning rate.

---

1 point

3.
Suppose you have $m = 23$ training examples with $n = 5$ features (excluding the additional all-ones feature for the intercept term, which you should add). The normal equation is $\theta = (X^T X)^{-1} X^T y$. For the given values of m and n, what are the dimensions of θ, X, and y in this equation?

- X is 23×5, y is 23×1, θ is 5×1
- X is 23×6, y is 23×1, θ is 6×1
- X is 23×5, y is 23×1, θ is 5×5
- X is 23×6, y is 23×6, θ is 6×6

1

point

4.

Suppose you have a dataset with $m = 50$ examples and $n = 15$ features for each example. You want to use multivariate linear regression to fit the parameters θ to our data. Should you prefer gradient descent or the normal equation?

- The normal equation, since it provides an efficient way to directly find the solution.
- Gradient descent, since $(X^T X)^{-1}$ will be very slow to compute in the normal equation.
- Gradient descent, since it will always converge to the optimal θ.
- The normal equation, since gradient descent might be unable to find the optimal θ.

---

1 point

5.

Which of the following are reasons for using feature scaling?

- [ ] It is necessary to prevent gradient descent from getting stuck in local optima.
- [ ] It prevents the matrix $X^T X$ (used in the normal equation) from being non-invertable (singular/degenerate).
- [ ] It speeds up solving for θ using the normal equation.
- [ ] It speeds up gradient descent by making it require fewer iterations to get to a good solution.

---

4 questions unanswered

Submit Quiz

Logistic Regression

5 questions

1 point

1.

Suppose that you have trained a logistic regression classifier, and it outputs on a new example x a prediction $h_\theta(x)$ = 0.7. This means (check all that apply):

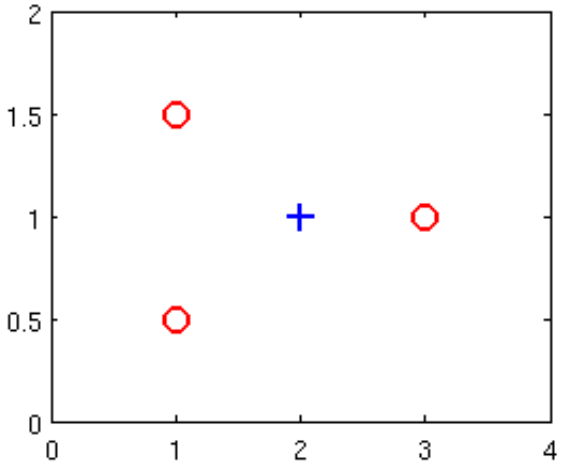
- ☐ Our estimate for $P(y=0|x;\theta)$ is 0.7.
- ☐ Our estimate for $P(y=1|x;\theta)$ is 0.3.
- ☐ Our estimate for $P(y=0|x;\theta)$ is 0.3.
- ☐ Our estimate for $P(y=1|x;\theta)$ is 0.7.

1 point

2.

Suppose you have the following training set, and fit a logistic regression classifier $h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$.

x_1	x_2	y
1	0.5	0
1	1.5	0
2	1	1
3	1	0



Which of the following are true? Check all that apply.

- ☐ Adding polynomial features (e.g., instead using $h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_1 x_2 + \theta_5 x_2^2)$) could increase how well we can fit the training data.
- ☐ At the optimal value of θ (e.g., found by fminunc), we will have $J(\theta) \geq 0$.
- ☐ Adding polynomial features (e.g., instead using $h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_1 x_2 + \theta_5 x_2^2)$) would increase $J(\theta)$ because we are now summing over more terms.
- ☐ If we train gradient descent for enough iterations, for some examples $x^{(i)}$ in the training set it is possible to obtain $h_\theta(x^{(i)}) > 1$.

1 point

3.

For logistic regression, the gradient is given by $\frac{\partial}{\partial \theta_j} J(\theta) = \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_j^{(i)}$. Which of these is a correct gradient descent update for logistic regression with a learning rate of α? Check all that apply.

- ☐ $\theta := \theta - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(\theta^T x - y^{(i)} \right) x^{(i)}$.
- ☐ $\theta_j := \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_j^{(i)}$ (simultaneously update for all j).
- ☐ $\theta_j := \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(\frac{1}{1+e^{-\theta^T x^{(i)}}} - y^{(i)} \right) x_j^{(i)}$ (simultaneously update for all j).
- ☐ $\theta_j := \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x^{(i)}$ (simultaneously update for all j).

---

1 point

4.

Which of the following statements are true? Check all that apply.

- ☐ The cost function $J(\theta)$ for logistic regression trained with $m \geq 1$ examples is always greater than or equal to zero.
- ☐ For logistic regression, sometimes gradient descent will converge to a local minimum (and fail to find the global minimum). This is the reason we prefer more advanced optimization algorithms such as fminunc (conjugate gradient/BFGS/L-BFGS/etc).
- ☐ Linear regression always works well for classification if you classify by using a threshold on the prediction made by linear regression.
- ☐ The sigmoid function $g(z) = \frac{1}{1+e^{-z}}$ is never greater than one (> 1).

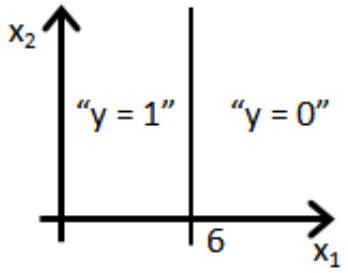
---

1 point

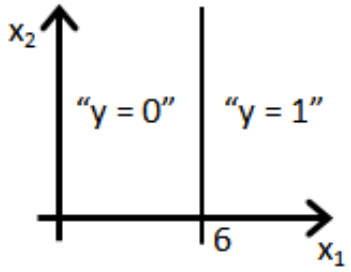
5.

Suppose you train a logistic classifier $h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$. Suppose $\theta_0 = -6, \theta_1 = 0, \theta_2 = 1$. Which of the following figures represents the decision boundary found by your classifier?

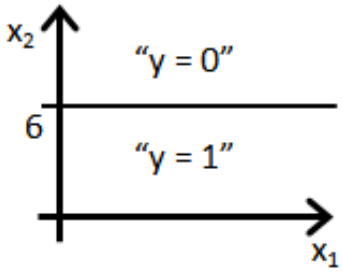
- Figure:



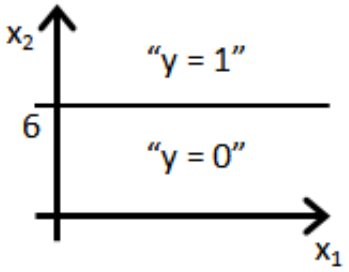
- Figure:



- Figure:



- Figure:



1 question unanswered

Submit Quiz

Regularization

5 questions

1 point

1.
You are training a classification model with logistic

regression. Which of the following statements are true? Check

all that apply.

- ☐ Introducing regularization to the model always results in equal or better performance on examples not in the training set.
- ☐ Introducing regularization to the model always results in equal or better performance on the training set.
- ☐ Adding a new feature to the model always results in equal or better performance on examples not in the training set.
- ☐ Adding many new features to the model makes it more likely to overfit the training set.

1 point

2.

Suppose you ran logistic regression twice, once with $\lambda = 0$, and once with $\lambda = 1$. One of the times, you got

parameters $\theta = \begin{bmatrix} 26.29 \\ 65.41 \end{bmatrix}$, and the other time you got

$\theta = \begin{bmatrix} 2.75 \\ 1.32 \end{bmatrix}$. However, you forgot which value of

λ corresponds to which value of θ. Which one do you

think corresponds to $\lambda = 1$?

- $\theta = \begin{bmatrix} 26.29 \\ 65.41 \end{bmatrix}$
- $\theta = \begin{bmatrix} 2.75 \\ 1.32 \end{bmatrix}$

---

1 point

3.
Which of the following statements about regularization are

true? Check all that apply.

- ☐ Because logistic regression outputs values $0 \leq h_\theta(x) \leq 1$, it's range of output values can only be "shrunk" slightly by regularization anyway, so regularization is generally not helpful for it.
- ☐ Using a very large value of λ cannot hurt the performance of your hypothesis; the only reason we do not set λ to be too large is to avoid numerical problems.
- ☐ Consider a classification problem. Adding regularization may cause your classifier to incorrectly classify some training examples (which it had correctly classified when not using regularization, i.e. when $\lambda = 0$).
- ☐ Using too large a value of λ can cause your hypothesis to overfit the data; this can be avoided by reducing λ.

---

1 point

4.

In which one of the following figures do you think the hypothesis has overfit the training set?

Figure:

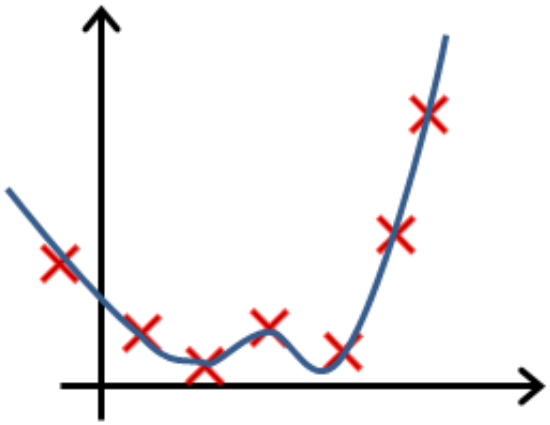


Figure:

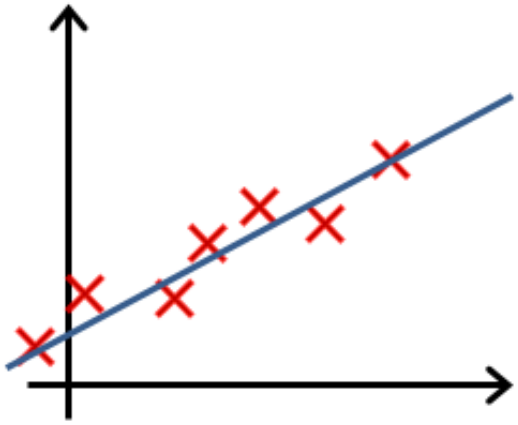


Figure:

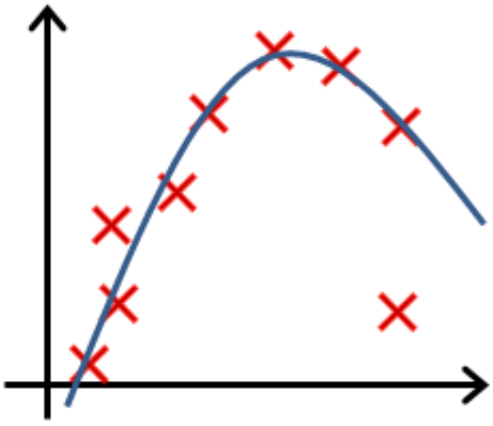
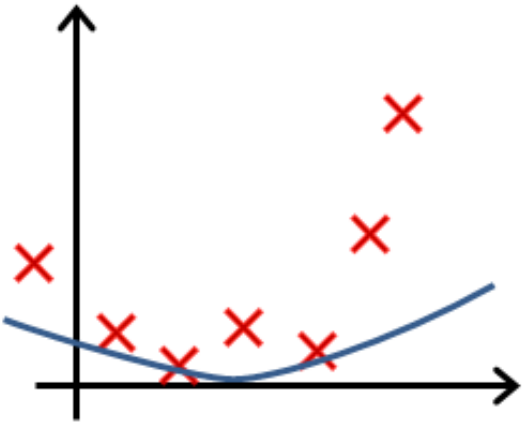


Figure:



1 point

5.
In which one of the following figures do you think the hypothesis has underfit the training set?

Figure:

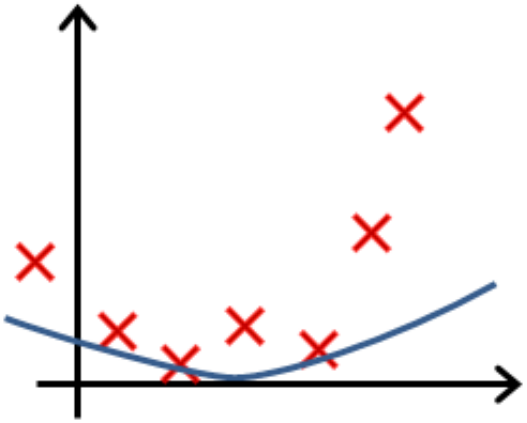


Figure:

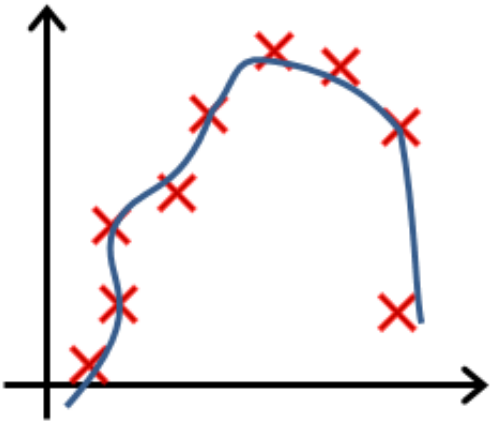


Figure:

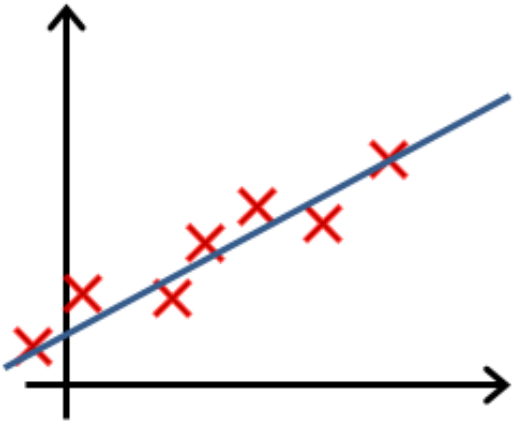
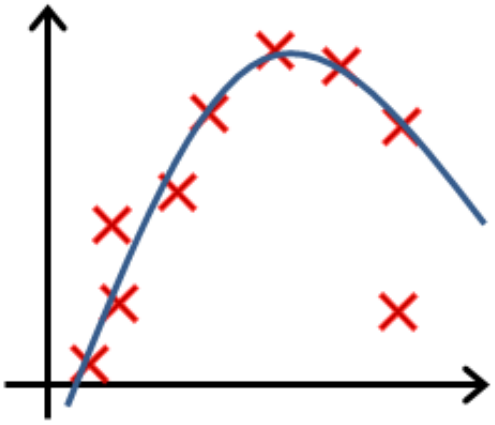


Figure:



3 questions unanswered

Submit Quiz

Unsupervised Learning

5 questions

1 point

1.
For which of the following tasks might K-means clustering be a suitable algorithm? Select all that apply.

- ☐ Given a database of information about your users, automatically group them into different market segments.
- ☐ Given sales data from a large number of products in a supermarket, figure out which products tend to form coherent groups (say are frequently purchased together) and thus should be put on the same shelf.
- ☐ Given historical weather records, predict the amount of rainfall tomorrow (this would be a real-valued output)
- ☐ Given sales data from a large number of products in a supermarket, estimate future sales for each of these products.

1 point

2.
Suppose we have three cluster centroids $\mu_1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$, $\mu_2 = \begin{bmatrix} -3 \\ 0 \end{bmatrix}$ and $\mu_3 = \begin{bmatrix} 4 \\ 2 \end{bmatrix}$. Furthermore, we have a training example $x^{(i)} = \begin{bmatrix} 3 \\ 1 \end{bmatrix}$. After a cluster assignment step, what will $c^{(i)}$ be?

- $c^{(i)}$ is not assigned
- $c^{(i)} = 3$
- $c^{(i)} = 2$
- $c^{(i)} = 1$

---

1 point

3.
K-means is an iterative algorithm, and two of the following steps are repeatedly carried out in its inner-loop. Which two?

- [ ] The cluster assignment step, where the parameters $c^{(i)}$ are updated.
- [ ] Move the cluster centroids, where the centroids μ_k are updated.
- [ ] Feature scaling, to ensure each feature is on a comparable scale to the others.
- [ ] Using the elbow method to choose K.

---

1 point

4.
Suppose you have an unlabeled dataset $\{x^{(1)}, \ldots, x^{(m)}\}$. You run K-means with 50 different random

initializations, and obtain 50 different clusterings of the

data. What is the recommended way for choosing which one of

these 50 clusterings to use?

- The answer is ambiguous, and there is no good way of choosing.

- For each of the clusterings, compute $\frac{1}{m}\sum_{i=1}^{m}||x^{(i)}-\mu_{c^{(i)}}||^2$, and pick the one that minimizes this.
- Always pick the final (50th) clustering found, since by that time it is more likely to have converged to a good solution.
- The only way to do so is if we also have labels $y^{(i)}$ for our data.

1 point

5.
Which of the following statements are true? Select all that apply.

- [ ] If we are worried about K-means getting stuck in bad local optima, one way to ameliorate (reduce) this problem is if we try using multiple random initializations.
- [ ] The standard way of initializing K-means is setting $\mu_1 = \cdots = \mu_k$ to be equal to a vector of zeros.
- [ ] Since K-Means is an unsupervised learning algorithm, it cannot overfit the data, and thus it is always better to have as large a number of clusters as is computationally feasible.
- [ ] For some datasets, the "right" or "correct" value of K (the number of clusters) can be ambiguous, and hard even for a human expert looking carefully at the data to decide.

2 questions unanswered

Submit Quiz